

Федеральное государственное бюджетное образовательное учреждение высшего образования



«Московский государственный технический
университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ФУНДАМЕНТАЛЬНЫЕ НАУКИ

КАФЕДРА МАТЕМАТИКА И КОМПЬЮТЕРНЫЕ НАУКИ

Отчет


по лабораторному заданию № 15

Вариант 18

Дисциплина: Информатика

Название лабораторного задания: Файлы. Текстовые файлы. Передача файлов в подпрограммы.

Студент гр. ФН11-22Б

 26.04.21

М.Х. Хаписов
(И.О. Фамилия)

Преподаватель Доцент кафедры ИУ-6


(Подпись, дата)

Т.Н. Ничушкина
(И.О. Фамилия)

26.04.2021

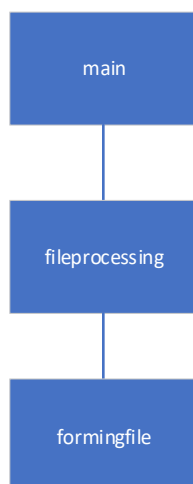
Москва, 2021

Файлы. Текстовые файлы. Передача файлов в подпрограммы.

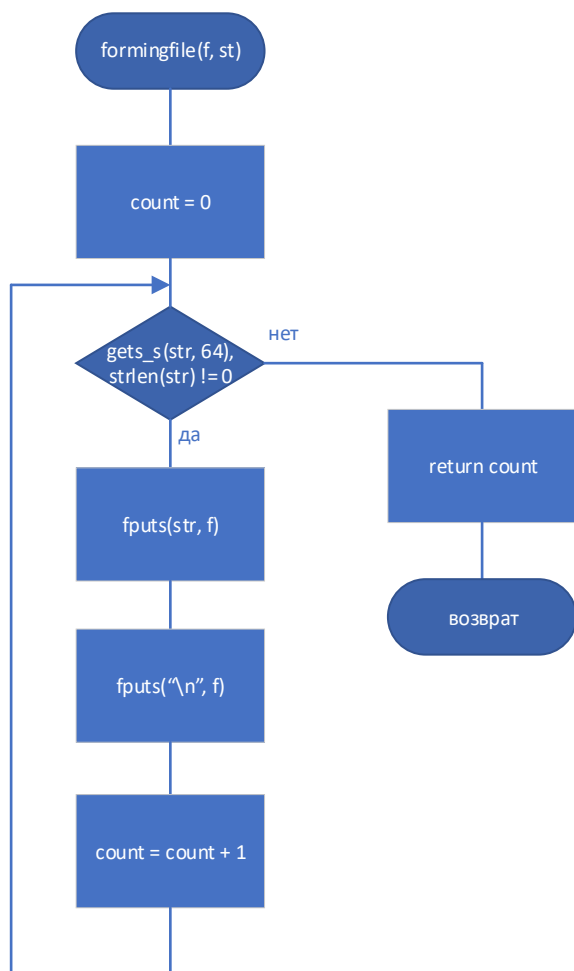
Цель: научиться работать с файлами и передавать их в подпрограммы в качестве параметров.

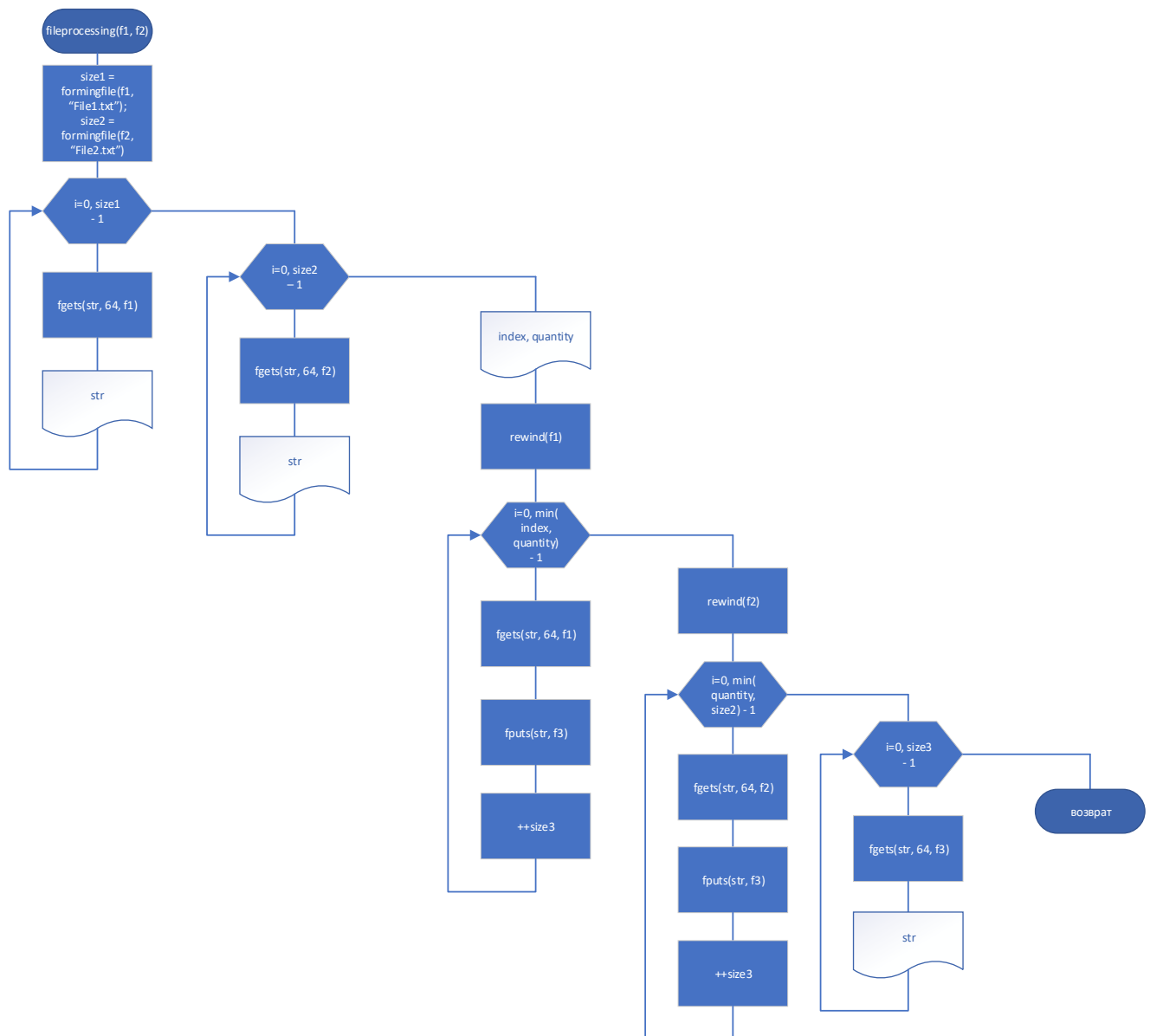
Задание А. Сформировать программным путем текстовый файл символьных строк. В полученный файл, после строки, номер которой введен с клавиатуры, вставить указанное количество строк из другого текстового файла. Проконтролировать значения номера вводимой строки и копируемого количества строк. Если номер строки выходит за пределы файла – выдать соответствующее сообщение и дописать нужные строки в конец файла. Если количество вставляемых строк больше размера второго файла – в исходный файл вставить весь второй файл. Вывести на экран сформированный и полученный файлы и всю необходимую информацию с комментариями.

Структурная схема



Схемы алгоритмов





Текст программы

```

#include <iostream>
int formingfile(FILE *f, const char* st) {
    std::cout << "\n----- Forming the file ----- \n";
    int count = 0;
    if (f != NULL) fclose(f);
    fopen_s(&f, st, "w+t");
    if (f != NULL) {
        char* str = new char[64];
        while (gets_s(str, 64), strlen(str) != 0) {
            fputs(str, f);
            fputs("\n", f);
            ++count;
        }
        delete[] str;
        fclose(f);
    }
    else std::cout << "Error! The file couldn't be opened\n";
    return count;
}

```

```

void fileprocessing(FILE *f1, FILE *f2) {
    if (f1 != NULL) fclose(f1);
    fopen_s(&f1, "File1.txt", "rt");
    int size1 = formingfile(f1, "File1.txt");
    if (f2 != NULL) fclose(f2);
    fopen_s(&f2, "File2.txt", "rt");
    int size2 = formingfile(f2, "File2.txt");

    fopen_s(&f1, "File1.txt", "rt");
    fopen_s(&f2, "File2.txt", "rt");
    FILE* f3;
    fopen_s(&f3, "File3.txt", "w+t");
    if (f1 != NULL && f2 != NULL && f3 != NULL) {
        char* str = new char[64];

        std::cout << "\n----- The first file ----- \n";
        for (int i = 0; i < size1; ++i) {
            fgets(str, 64, f1);
            std::cout << str;
        }

        std::cout << "\n----- The second file ----- \n";
        for (int i = 0; i < size2; ++i) {
            fgets(str, 64, f2);
            std::cout << str;
        }

        int index, quantity;
        std::cout << "\nInput the index and the quantity of the pasted strings from the
second file\n";
        std::cin >> index >> quantity;

        int size3 = 0;
        rewind(f1);
        for (int i = 0; i < index && i < size1; ++i) {
            fgets(str, 64, f1);
            fputs(str, f3);
            ++size3;
        }

        if (size1 < index)
            std::cout << "\nThe index was more than the size of the first file\n";

        rewind(f2);
        for (int i = 0; i < quantity && i < size2; ++i) {
            fgets(str, 64, f2);
            fputs(str, f3);
            ++size3;
        }

        if (size2 < quantity)
            std::cout << "\nThe quantity was more than the size of the second file\n";

        std::cout << "\n----- The transformed file ----- \n";
        fclose(f3);
        fopen_s(&f3, "File3.txt", "rt");
        if (f3 != NULL) {
            for (int i = 0; i < size3; ++i) {
                fgets(str, 64, f3);
                std::cout << str;
            }
            fclose(f3);
        }
    }
}

```

```

        }
        fclose(f1);
        fclose(f2);
    }
    else std::cout << "Error! The files couldn't be opened\n";
    remove("File1.txt");
    remove("File2.txt");
    remove("File3.txt");
}

void main() {
    FILE *f1;
    fopen_s(&f1, "File1.txt", "w+b");
    if (f1 != NULL) fclose(f1);

    FILE* f2;
    fopen_s(&f2, "File2.txt", "w+b");
    if (f2 != NULL) fclose(f2);

    fileprocessing(f1, f2);
    system("pause");
}

```

Тестирование

C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe

----- Forming the file -----

1 2 3
4 5 6
7 8 9

----- Forming the file -----

a b c d
e f g h

----- The first file -----

1 2 3
4 5 6
7 8 9

----- The second file -----

a b c d
e f g h

Input the index and the quantity of the pasted strings from the second file

4 3

The index was more than the size of the first file

The quantity was more than the size of the second file

----- The transformed file -----

1 2 3
4 5 6
7 8 9
a b c d
e f g h

Для продолжения нажмите любую клавишу . . .

C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe

----- Forming the file -----

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

----- Forming the file -----

a b c
d e f
g h i
j k l
m n o
p q r
s t u
v w x
y z

----- The first file -----

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

----- The second file -----

a b c
d e f
g h i
j k l
m n o
p q r
s t u
v w x
y z

Input the index and the quantity of the pasted strings from the second file

4 3

----- The transformed file -----

1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
a b c
d e f
g h i

Для продолжения нажмите любую клавишу . . .


```

C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe
Input the index and the quantity of the pasted strings from the second file
4 3

----- Forming the file -----
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

----- Forming the file -----
a b c
d e f
g h i
j k l
m n o
p q r
s t u
v w x
y z

----- The first file -----
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

----- The second file -----
a b c
d e f
g h i
j k l
m n o
p q r
s t u
v w x
y z

----- The transformed file -----
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
a b c
d e f
g h i
Для продолжения нажмите любую клавишу . . .

```

Вывод : я научился работать с текстовыми файлами и передавать их в подпрограммы

Задание Б. Записать в файл введенную последовательность вещественных чисел. Просмотреть файл и удалить компоненты равные 1-му, 2-му и 3-му (сами компоненты с указанными номерами в файле оставить). Вывести на экран исходный и полученный файлы.

Структурная схема

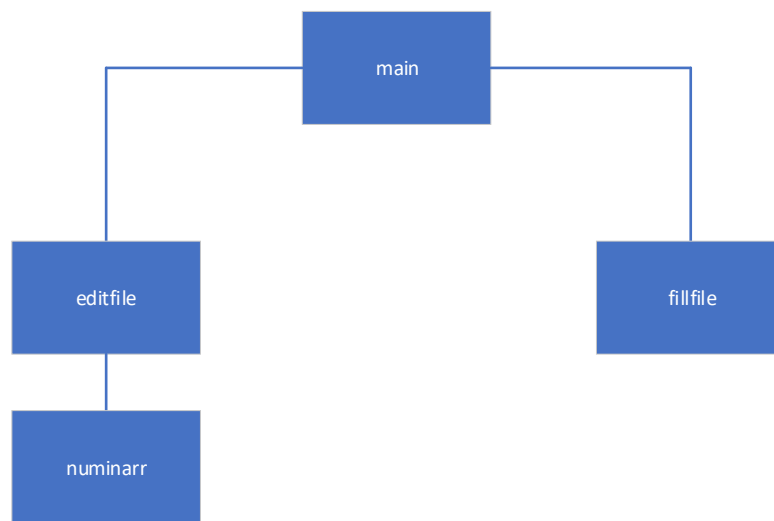
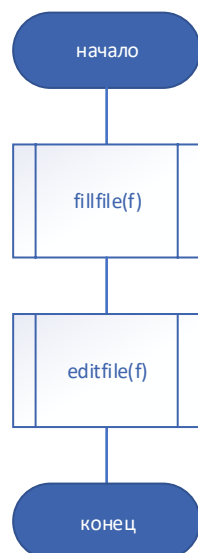
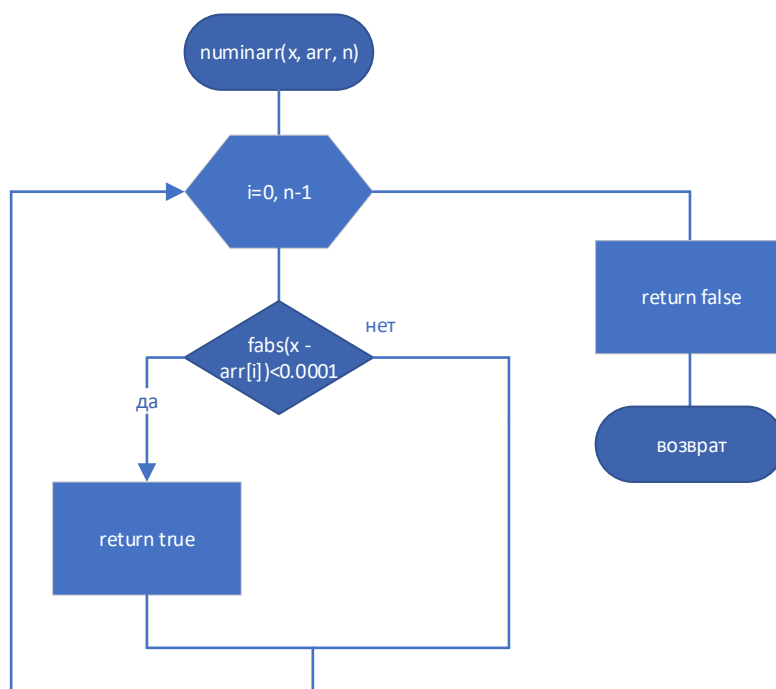
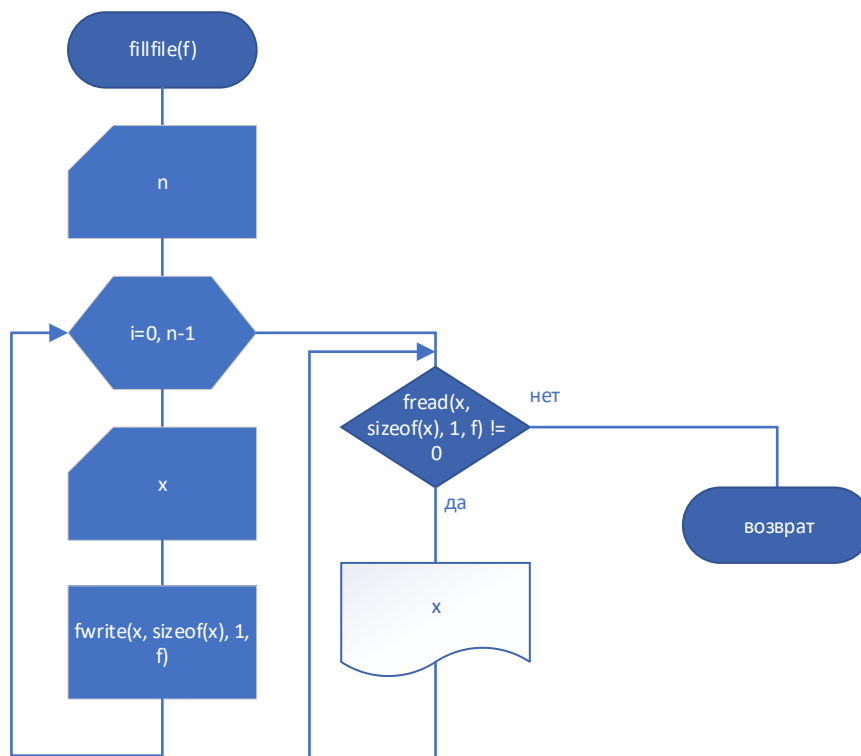
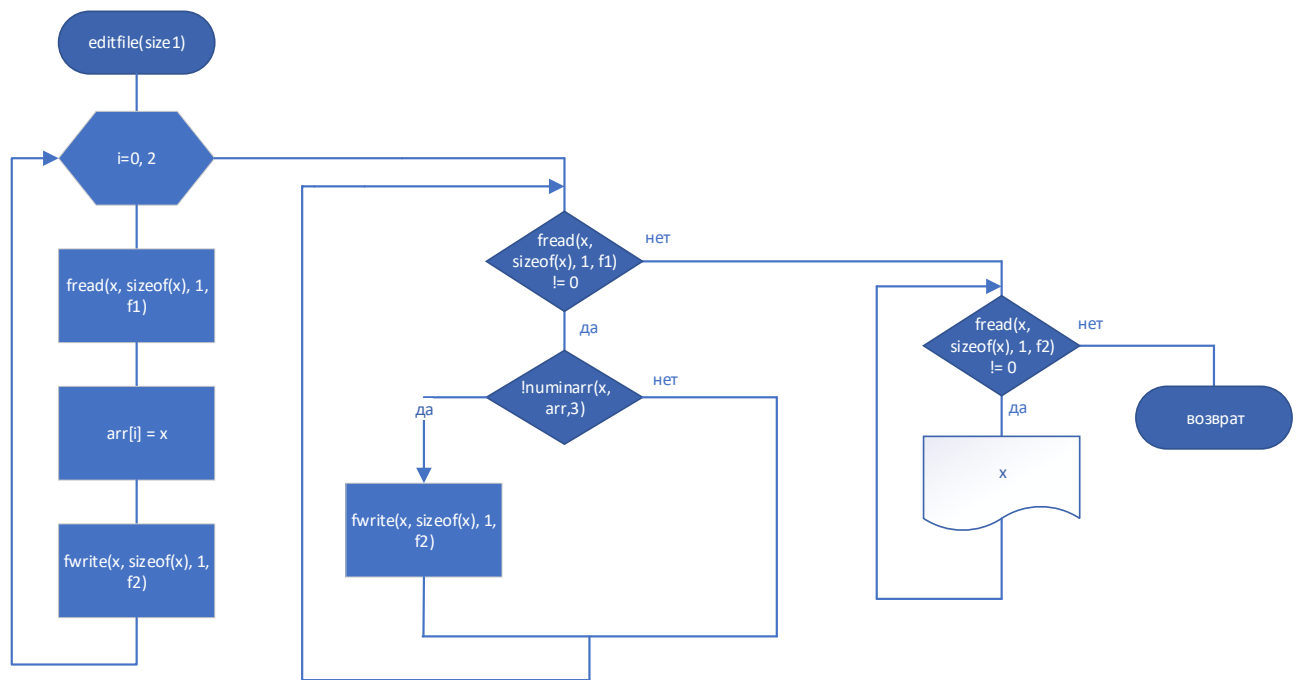


Схема алгоритмов







Текст программы

```
#include <iostream>
#include <cmath>
```

```
void fillfile(FILE* f) {
    int n;
    std::cout << "Input the quantity of numbers in the sequence\n";
    std::cin >> n;
    while (n < 3) {
        std::cout << "The number cannot be less 3\n";
        std::cin >> n;
    }

    if (f != NULL) fclose(f);
    fopen_s(&f, "File1.bin", "w+b");
    if (f != NULL) {
        std::cout << "Input the sequence of real numbers\n";
        for (int i = 0; i < n; ++i) {
            double x;
            std::cin >> x;
            fwrite(&x, sizeof(x), 1, f);
        }
        fclose(f);
    }

    std::cout << "----- The initial file ----- \n";
    fopen_s(&f, "File1.bin", "rb");
    if (f != NULL) {
        double x;
        while (fread(&x, sizeof(x), 1, f) != 0)
            std::cout << x << " ";
    }
}
```

```

        std::cout << std::endl;
        fclose(f);
    }
}

bool numinarr(double x, double* arr, int n) {
    for (int i = 0; i < n; ++i) {
        if (fabs(x - *(arr + i)) < 0.0001)
            return true;
    }
    return false;
}

void editfile(FILE* f1) {
    if (f1 != NULL) fclose(f1);
    fopen_s(&f1, "File1.bin", "rb");
    FILE* f2;
    fopen_s(&f2, "File2.bin", "w+b");

    if (f1 != NULL && f2 != NULL) {
        double x;
        double* arr = new double[3];

        for (int i = 0; i < 3; ++i) {
            fread(&x, sizeof(x), 1, f1);
            *(arr + i) = x;
            fwrite(&x, sizeof(x), 1, f2);
        }

        std::cout << "\nDeleted numbers: ";

        while (fread(&x, sizeof(x), 1, f1) != 0) {
            if (!numinarr(x, arr, 3))
                fwrite(&x, sizeof(x), 1, f2);
            else
                std::cout << x << " ";
        }

        std::cout << std::endl << std::endl;

        delete[] arr;
        fclose(f1);
        remove("File1.bin");
        fclose(f2);
        fopen_s(&f2, "File2.bin", "rb");

        std::cout << "----- The transformed file -----
\n";

        if (f2 != NULL) {
            double x;
            while (fread(&x, sizeof(x), 1, f2) != 0)
                std::cout << x << " ";
            std::cout << std::endl;
            fclose(f2);
        }
        remove("File2.bin");
    }
}

void main() {
    FILE* f;
    fopen_s(&f, "File1.bin", "w+b");

```

```

        if (f != NULL)
            fclose(f);
        fillfile(f);
        editfile(f);
        system("pause");
    }

```

Тестирование

```

C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe
Input the quantity of numbers in the sequence
10
Input the sequence of real numbers
1 2 3 4 5 5 4 3 2 1
----- The initial file -----
1 2 3 4 5 5 4 3 2 1

Deleted numbers: 3 2 1

----- The transformed file -----
1 2 3 4 5 5 4
Для продолжения нажмите любую клавишу . . .

```

```

C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe
Input the quantity of numbers in the sequence
2
The number cannot be less 3
1
The number cannot be less 3
0
The number cannot be less 3
5
Input the sequence of real numbers
0 1 2 1 0
----- The initial file -----
0 1 2 1 0

Deleted numbers: 1 0

----- The transformed file -----
0 1 2
Для продолжения нажмите любую клавишу . . .

```

```
C:\Users\Knigan\source\repos\Project1\Debug\Project1.exe
Input the quantity of numbers in the sequence
5
Input the sequence of real numbers
-1.2 2.3 5.7 -4.6 1.8
----- The initial file -----
-1.2 2.3 5.7 -4.6 1.8

Deleted numbers:

----- The transformed file -----
-1.2 2.3 5.7 -4.6 1.8
Для продолжения нажмите любую клавишу . . .
```

Вывод 2: я научился работать с двоичными файлами и передавать их в подпрограммы

Ответы на контрольные вопросы

1. Что такое файл, и какие типы файлов вы знаете? (Ответ: файл – поименованная последовательность элементов данных (компонент файла))

2. Что такое файловая переменная и как ее объявить в программе на C++? (Ответ: файловая переменная – это указатель на файл)

FILE *<Файловая переменная>

3. Как открыть файл на чтение или запись с контролем наличия файла на диске? (Ответ: добавить к операции +)

Приведите пример.

f = fopen("File.txt", "w"); //файл открыт на запись, причём, если файла не существует, fopen вернёт NULL

f = fopen("File.txt", "w+"); //файл открыт на запись, причём, если файла не существует, будет создан новый файл

4. Что такое файловый указатель? Каково его назначение? (Ответ: файловый указатель определяет, в какое место файла выводится запись или откуда она вводится)

5. Какие Вы знаете функции работы с файловым указателем? (Ответ: ftell, rewind, fseek)
6. Что такое стандартные текстовые файлы? Где они используются? (Ответ: Текстовый файл представляет собой последовательность символьных строк переменной длины, заканчивающихся маркером <<Конец строки>>)
7. Каковы особенности работы с текстовыми файлами? (Ответ: вместо функций fwrite и fread используются fputs и fgets)
8. Что такое форматный ввод-вывод? Приведите основные функции работы с ним? (Ответ: Операции форматного ввода/вывода предполагают, что при выполнении этих действий осуществляется: при вводе – преобразование символьного представления данных во внутреннее машинное представление в соответствии с заданным форматом и, наоборот, при выводе – преобразование из внутреннего формата в символьный. Основные функции: fprintf и fscanf)